Stackelberg Strategies

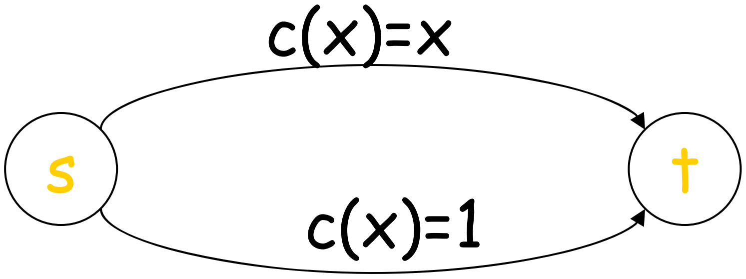
Algorithmic Game Theory Course

Co.RE.Lab. - N.T.U.A.

Stackelberg Routing

- In (classic) selfish routing all players act **selfishly**.
- In Stackelberg routing there exist players willing to **cooperate** for social welfare (a fraction of the total players).
  - Both Selfish and Cooperative players are present.
  - Leader determines the paths of the coordinated players.
  - Selfish players (followers) minimize their own cost.
- **Nash Equilibria** are considered as the possible outcomes of the game.
- A **Stackelberg Strategy** is an algorithm that allocates paths to coordinated players so as to lead selfish players to a good Nash Equilibrium.

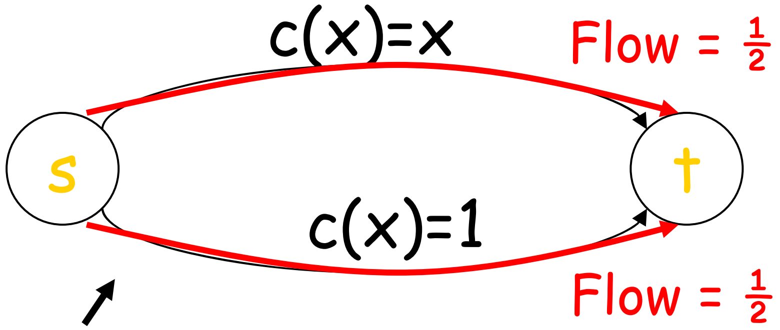
Example: Pigou's Network



One unit of flow is to be routed from s to t

Example: Pigou's Network

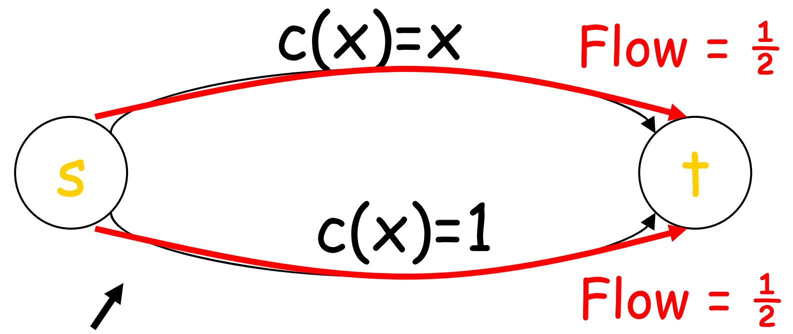
One unit of flow is to be routed from s to t



Optimal flow

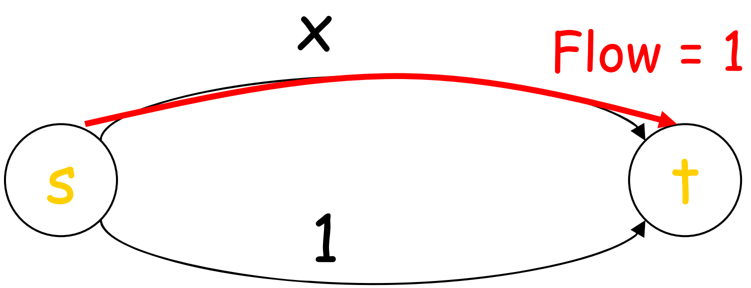
Example: Pigou's Network

One unit of flow is to be routed from s to t



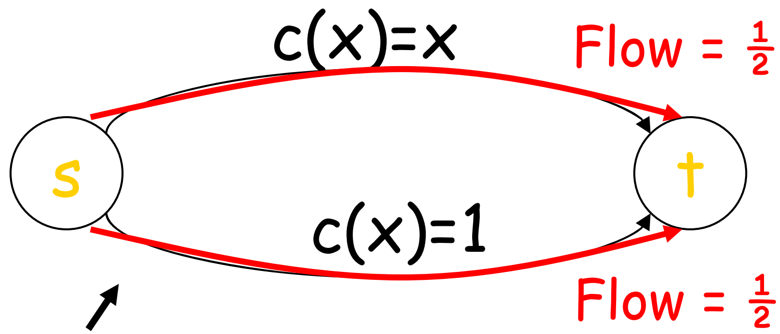
Optimal flow

(Classic) Nash flow



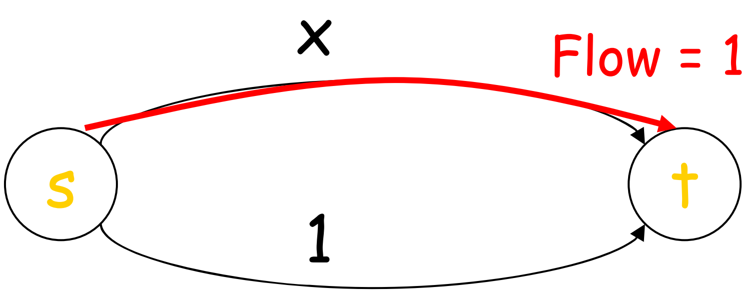
Example: Pigou's Network

One unit of flow is to be routed from s to t

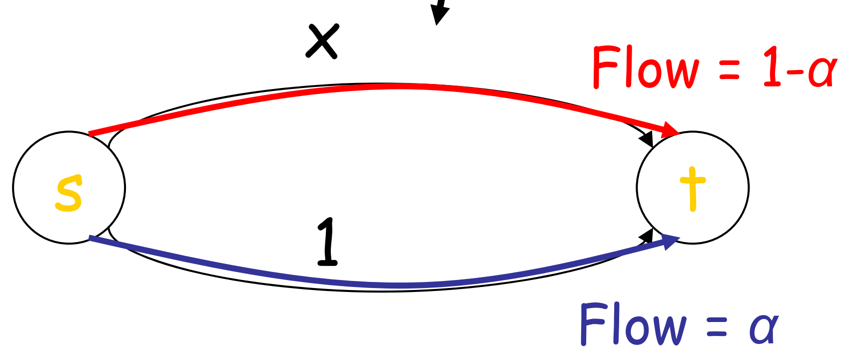


Optimal flow

(Classic) Nash flow

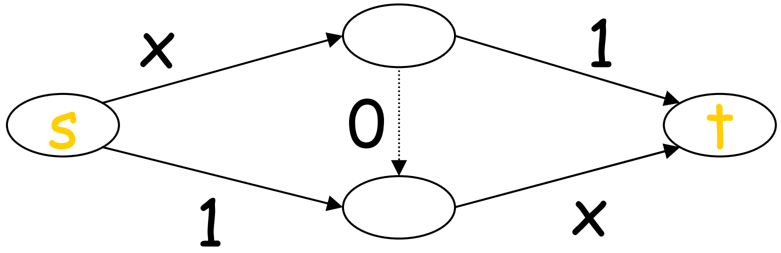


Nash flow when **a fraction *α* of (coordinated) players** <u>is sent</u> through the lower edge



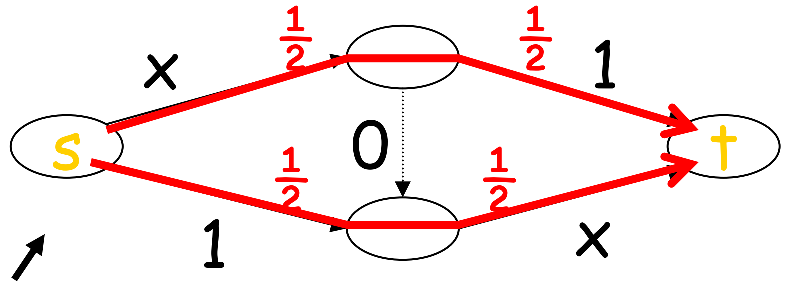
Example: Braess's Network

One unit of flow is to be routed from s to t



Example: Braess's Network

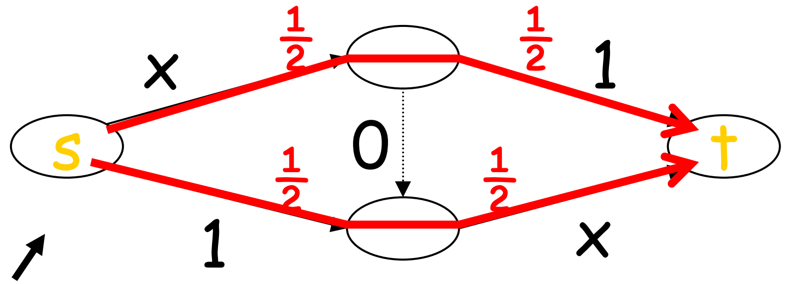
One unit of flow is to be routed from s to t



Optimal flow

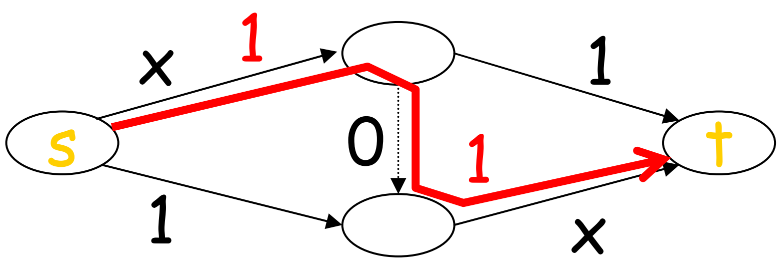
Example: Braess's Network

One unit of flow is to be routed from s to t



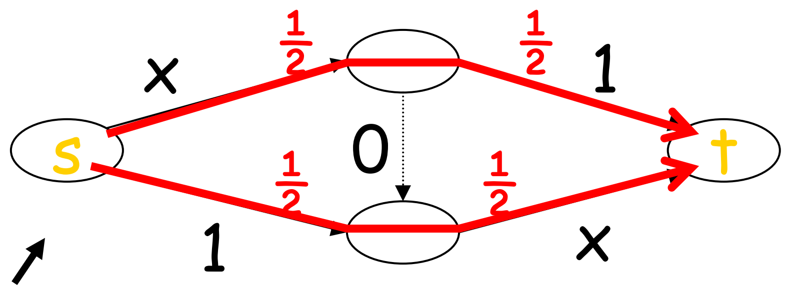
Optimal flow

(Classic) Nash flow



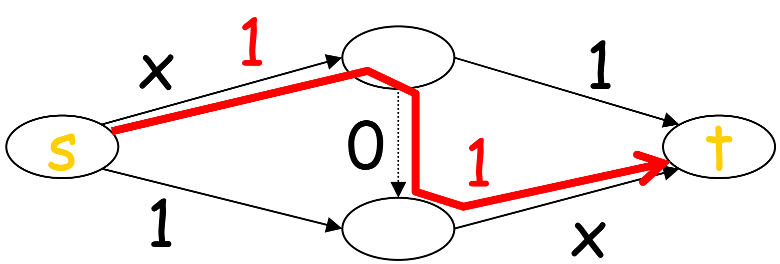
Example: Braess's Network

One unit of flow is to be routed from s to t

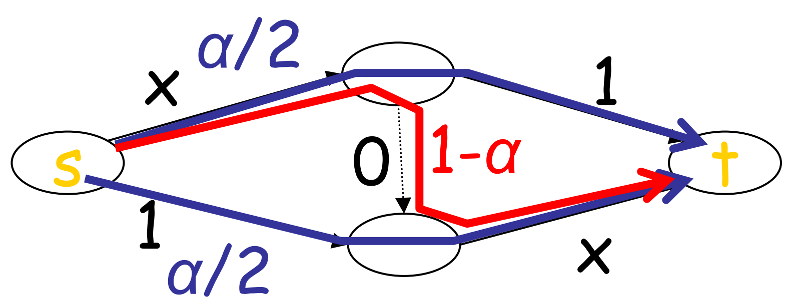


Optimal flow

(Classic) Nash flow



Nash flow when **a fraction *α* of coordinated players** is sent through the lower edge



Slightly more formal

- We will consider single commodity networks.
- An instance in such networks: (G, c_e, r)
- Assume that a fraction *α* of the players are cooperative. (G, c_e, r, α)
- A Stackelberg strategy assigns cooperative players to paths.
  - They induce a congestion $s = \{s_e\}_{e \in E}$
- A new game is "created": $(G, c'_e, (1-\alpha)r)$
  - Where $c'_e(x) = c_e(x + s_e)$

In the "new" game

- Selfish players choose paths (as usual), and Nash flows are considered as the possible outcomes of the game (as usual).
- On Equilibrium, selfish players induce a congestion $\sigma = \{\sigma_e\}_{e \in E}$
- The Price of Anarchy is $PoA = \dfrac{C(\sigma + s)}{OPT}$

The Central Questions

- Given a Stackelberg routing instance, we can ask:
  - Among all Stackelberg strategies, can we characterize and/or compute the strategy inducing the **Stackelberg equilibrium** - i.e., the eq. of minimum total latency?
  - What is the worst-case ratio between the total latency of the Stackelberg eq. and that of the optimal assignment of users to paths?

Finding best strategy: NP-hard

Reduction from $\frac{1}{3}$-$\frac{2}{3}$ $\mathrm{Partition}$ problem:

Given *n* positive integers $a_1, \ldots, a_n$ is there an $S \subseteq \{1, \ldots, n\}$ satisfying:

$$\sum_{i \in S} a_i = \frac{1}{3} \sum_{i=1}^{n} a_i$$

Given an instance of $\frac{1}{3}$-$\frac{2}{3}$ $\mathrm{Partition}$ create an instance of stackelberg routing:

- A network *G* with *n*+1 parallel links
- Demand: $2 \sum_{i=1}^{n} a_i = 2A$
- ¼ of the players are followers
- Cost functions: $c_i(x) = \frac{x}{a_i} + 4, i \leq n$ and $c_{n+1}(x) = \frac{x}{A}$

$$\text{"yes" instance} \Leftrightarrow \text{there exist a strategy with average cost} = \frac{35}{4} A$$

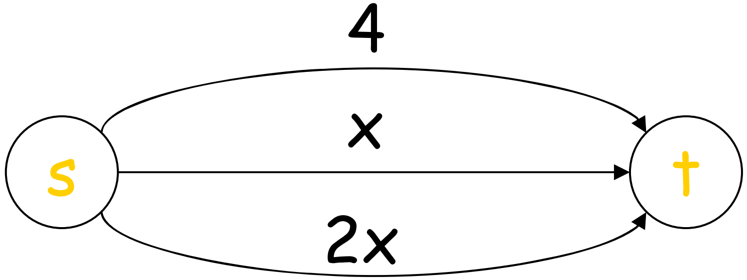
LLF Strategy

- Largest Latency First (LLF):
  - Compute an optimal configuration
  - Assign coordinated players to optimal paths of largest latency

LLF Strategy

- Largest Latency First (LLF):
  - Compute an optimal configuration
  - Assign coordinated players to optimal paths of largest latency

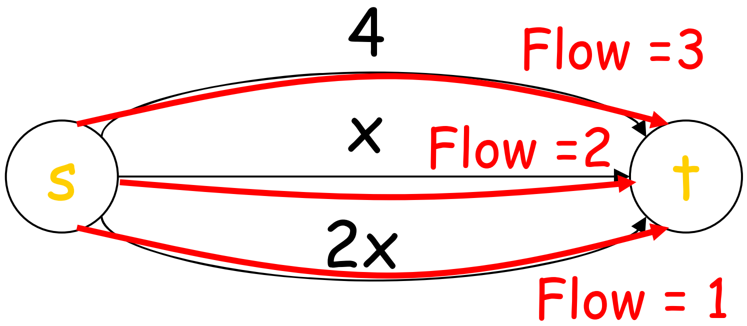
6 units to be routed.



LLF Strategy

- Largest Latency First (LLF):
  - Compute an optimal configuration
  - Assign coordinated players to optimal paths of largest latency

6 units to be routed.



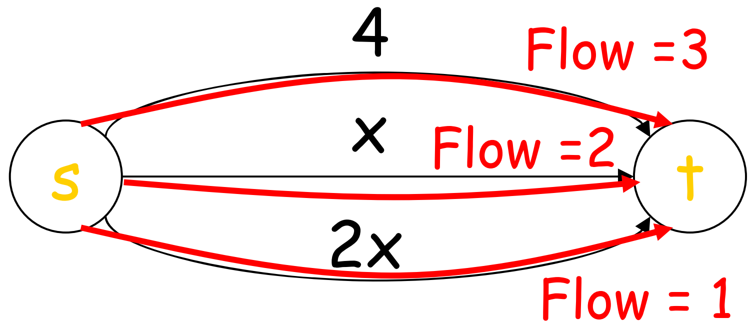
Opt routes:

- 3 to upper edge
- 2 to middle edge
- 1 to lower edge

LLF Strategy

- Largest Latency First (LLF):
  - Compute an optimal configuration
  - Assign coordinated players to optimal paths of largest latency

6 units to be routed.

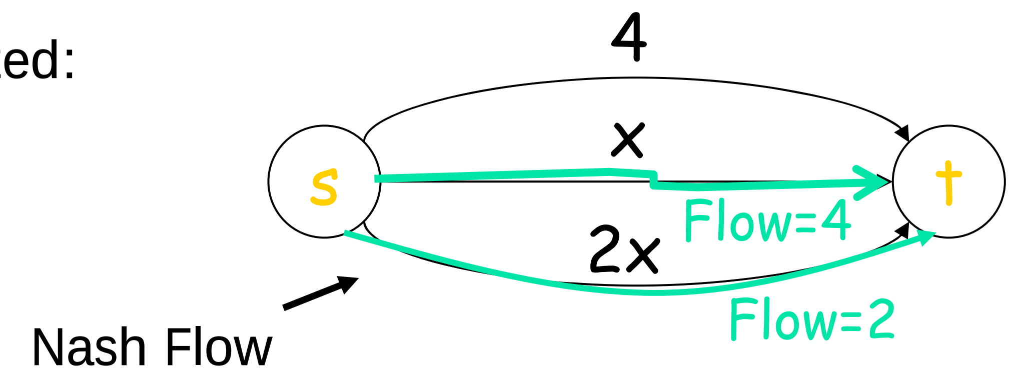


Opt routes:

- 3 to upper edge
- 2 to middle edge
- 1 to lower edge

In Nash Flow players are routed:

- 4 to middle edge
- 2 to lower edge



LLF Strategy

- Largest Latency First (LLF):
  - Compute an optimal configuration
  - Assign coordinated players to optimal paths of largest latency

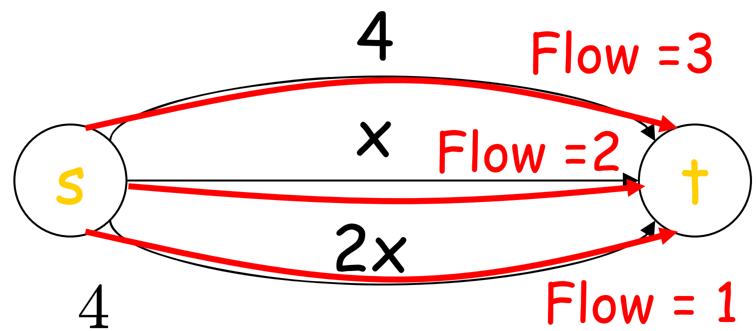
6 units to be routed.

Opt routes:

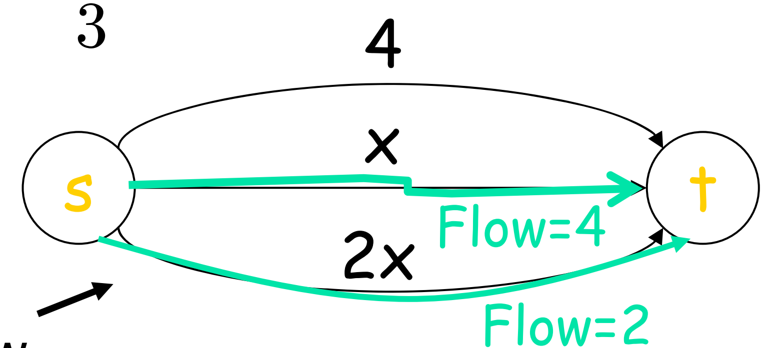
- 3 to upper edge
- 2 to middle edge
- 1 to lower edge

In Nash Flow players are routed:

- 4 to middle edge
- 2 to lower edge



$$PoA = \frac{4}{3}$$

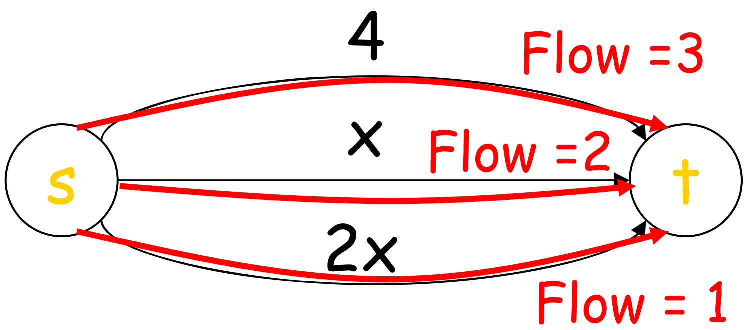


Nash Flow

LLF Strategy

- Largest Latency First (LLF):
  - Compute an optimal configuration
  - Assign coordinated players to optimal paths of largest latency

6 units to be routed.



Opt routes:

- 3 to upper edge
- 2 to middle edge
- 1 to lower edge

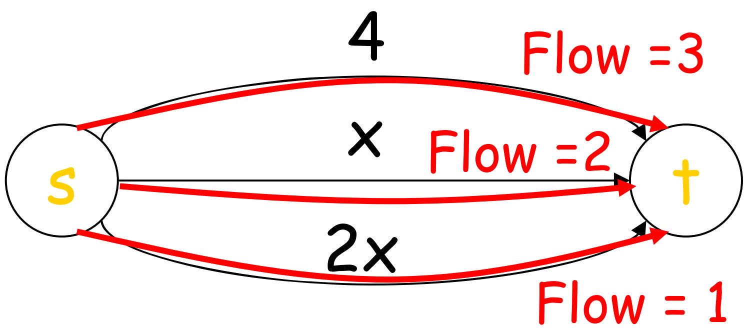
LLF Strategy

- Largest Latency First (LLF):
  - Compute an optimal configuration
  - Assign coordinated players to optimal paths of largest latency

6 units to be routed.

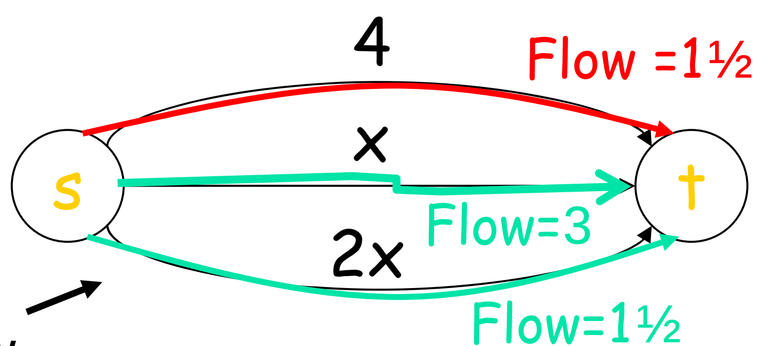
Opt routes:

- 3 to upper edge
- 2 to middle edge
- 1 to lower edge



LLF controlling ¼ players, e.g. 1½ units, routes:

- 1½ to upper edge

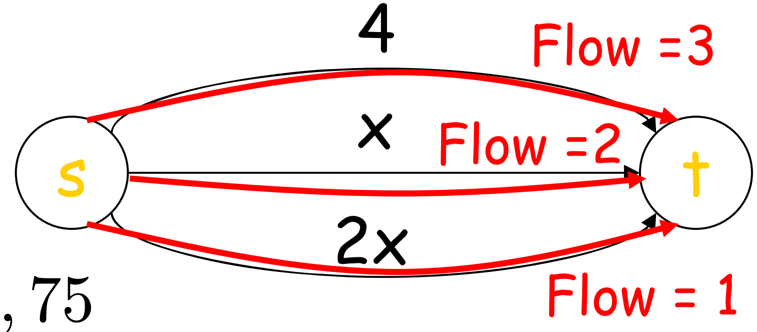


Nash Flow

LLF Strategy

- Largest Latency First (LLF):
  - Compute an optimal configuration
  - Assign coordinated players to optimal paths of largest latency

6 units to be routed.



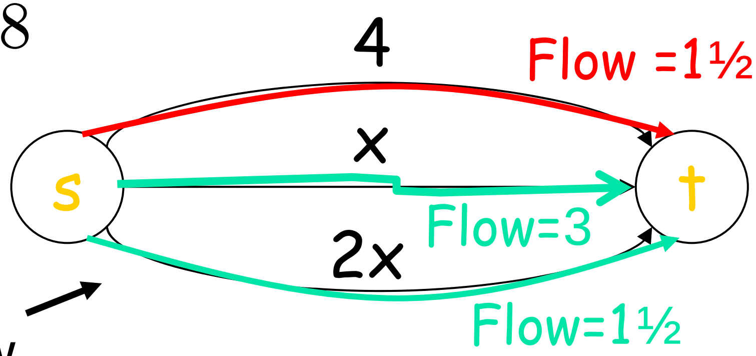
Opt routes:

- 3 to upper edge
- 2 to middle edge
- 1 to lower edge

$$PoA = \frac{18,75}{18}$$

LLF controlling ¼ players, e.g. 1½ units, routes:

- 1½ to upper edge



Nash Flow

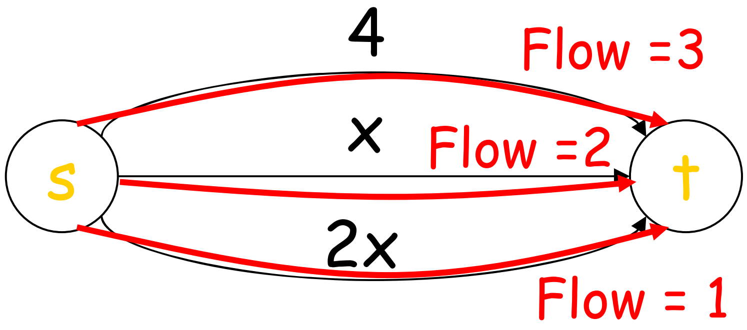
LLF Strategy

- Largest Latency First (LLF):
  - Compute an optimal configuration
  - Assign coordinated players to optimal paths of largest latency

6 units to be routed.

Opt routes:

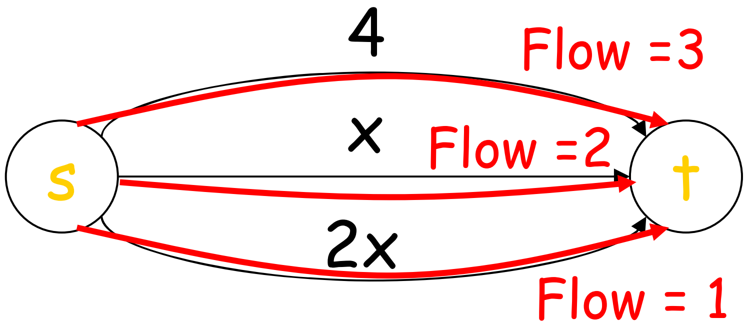
- 3 to upper edge
- 2 to middle edge
- 1 to lower edge



LLF Strategy

- Largest Latency First (LLF):
  - Compute an optimal configuration
  - Assign coordinated players to optimal paths of largest latency

6 units to be routed.

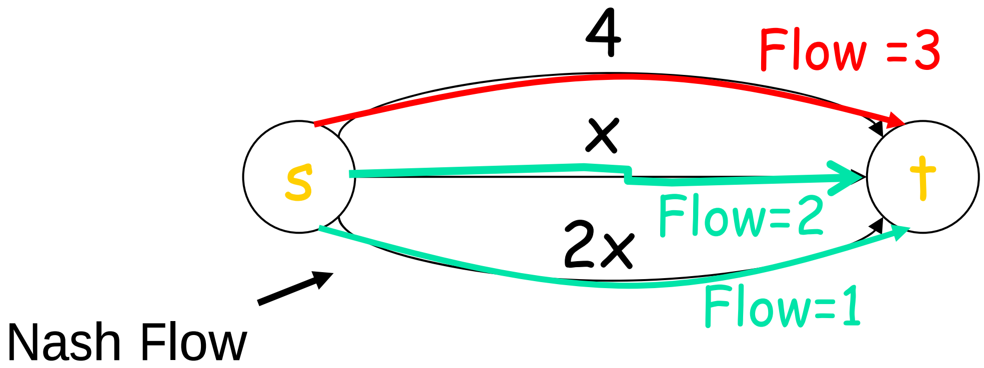


Opt routes:

- 3 to upper edge
- 2 to middle edge
- 1 to lower edge

LLF controlling ½ players,
e.g. 3 units, routes:

- 3 to upper edge



LLF Strategy

- Largest Latency First (LLF):
  - Compute an optimal configuration
  - Assign coordinated players to optimal paths of largest latency

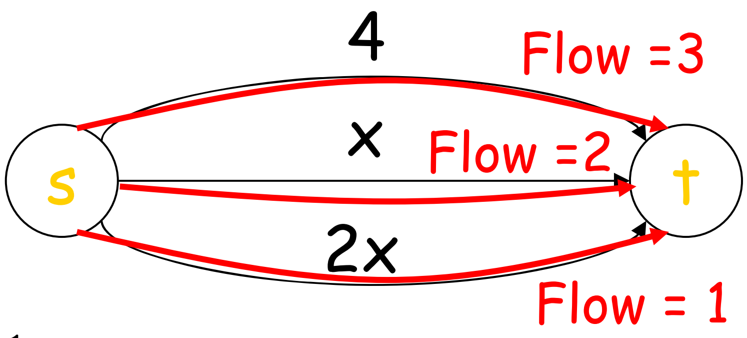
6 units to be routed.

Opt routes:

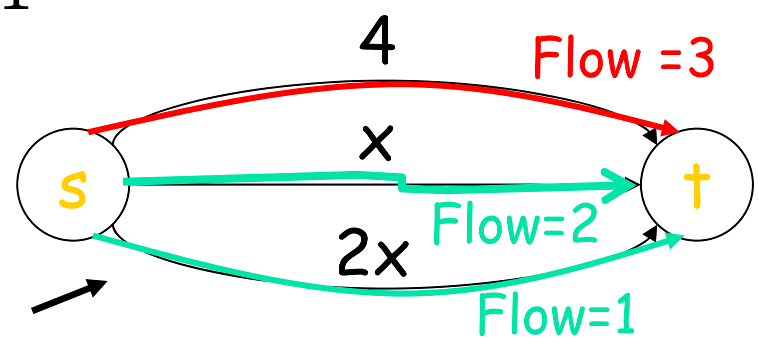
- 3 to upper edge
- 2 to middle edge
- 1 to lower edge

LLF controlling ½ players,
e.g. 3 units, routes:

- 3 to upper edge



$$PoA = 1$$



Nash Flow

LLF in parallel links

Let α be the fraction of the cooperative players.

Theorem 1: In parallel links LLF induces an assignment of cost no more than $1/\alpha$ times the OPT:

$$PoA_{LLF} \leq \frac{1}{\alpha}$$

Proof by induction: When LLF saturates a link we can restrict to the instance that has:

- this link deleted and
- fraction of players the “remainders” of the previous instance.

Some problems:

- LLF may fail to saturate any link. No problem: Let m be the “heaviest” link. If L is the cost of selfish players and x* is the optimal assignment, it is

$$OPT \geq x^* c_m(x_m^*) \geq \alpha L = \alpha C(s + \sigma)$$

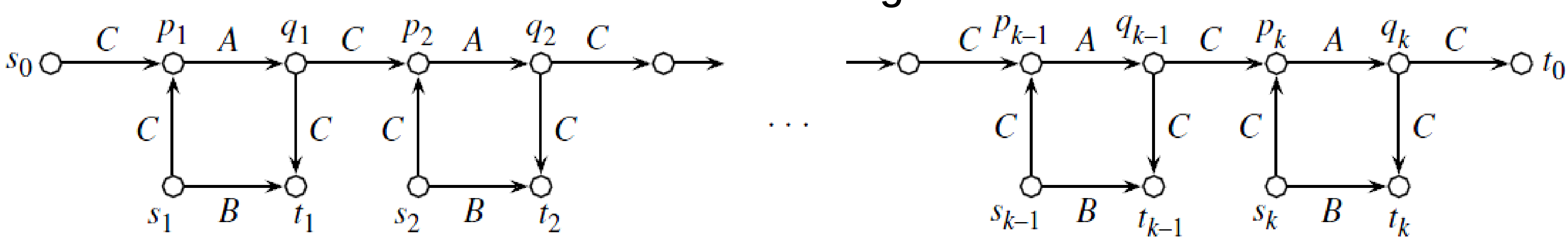
- When a link gets saturated selfish users could use it. No problem: There is an induced equilibrium that doesn’t use it.

Networks with Unbounded PoA

Theorem: Let $M > 0$ and $\alpha \in (0,1)$. There is an instance (G, c_e, r, α) such that for any Stackelberg strategy inducing *s*, it is:

$$C(s+\sigma) \geq M \cdot OPT$$

Proof: The network is the following



The demands are: $r_0 = \frac{1-\alpha}{2}$ and $r_i = \frac{1+\alpha}{2k}, i \geq 1$ (total flow=1)

Cost functions: B=1, C=0 and A is $c_\epsilon(x) = \begin{cases} 0, & \text{if } x \leq r_0; \\ 1 - \dfrac{r_0 + r_1 - x}{(1-\epsilon)r_1}, & \text{if } x \geq r_0 + 2\epsilon r_1. \end{cases}$

LLF in parallel links

Let o_e denote the optimal congestion

Lemma:

i) $C(s+\sigma) = \sum (s_e+\sigma_e) c_e(s_e+\sigma_e) \leq \rho \cdot OPT$

ii) $\sum \sigma_e c_e(s_e+\sigma_e) \leq \rho \cdot \sum (o_e - s_e) c_e(o_e)$

The proof follows from the variational inequality, similar to the "classic" result.

LLF in parallel links

Let o_e denote the optimal congestion

Lemma: i) $C(s+\sigma) = \sum (s_e + \sigma_e) c_e(s_e + \sigma_e) \le \rho \cdot OPT$

ii) $\sum \sigma_e c_e(s_e + \sigma_e) \le \rho \cdot \sum (o_e - s_e) c_e(o_e)$

The proof follows from the variational inequality, similar to the "classic" result.

Theorem 2: $PoA_{LLF} \le \alpha + (1-\alpha)\cdot\rho$

Proof: $OPT = \overbrace{\sum s_e c_e(o_e)}^{A} + \overbrace{\sum (o_e - s_e) c_e(o_e)}^{B}$ and $\frac{A}{B} \ge \frac{\alpha}{1-\alpha}$.

It is $C(s+\sigma) = \sum s_e c_e(s_e+\sigma_e) + \sum \sigma_e c_e(s_e + \sigma_e) \le A + \rho \cdot B$

This is maximized for $\frac{A}{B} = \frac{\alpha}{1-\alpha}$ with maximum value $\alpha + (1-\alpha)\cdot\rho$